Google Inc.

Titan Key

FIPS 140-2 Security Policy

Hardware version: 1.0

Firmware version: 1.0

Date: 09/21/2017

Prepared by:
Acumen Security
18504 Office Park Dr.
Montgomery Village, MD 20886
www.acumensecurity.net

Introduction

Federal Information Processing Standards Publication 140-2 — Security Requirements for Cryptographic Modules specifies requirements for cryptographic modules to be deployed in a Sensitive but Unclassified environment. The National Institute of Standards and Technology (NIST) and Communications Security Establishment Canada (CSE) Cryptographic Module Validation Program (CMVP) run the FIPS 140 program. The NVLAP accredits independent testing labs to perform FIPS 140 testing; the CMVP validates modules meeting FIPS 140 validation. Validated is the term given to a module that is documented and tested against the FIPS 140 criteria.

More information is available on the CMVP website at:
http://csrc.nist.gov/groups/STM/cmvp/index.html

About this Document

This non-proprietary Cryptographic Module Security Policy for Titan Key from Google Inc. provides an overview of the product and a high-level description of how it meets the overall Level 1 security requirements of FIPS 140-2.

Titan Key may also be referred to as the “module” in this document.

Disclaimer

The contents of this document are subject to revision without notice due to continued progress in methodology, design, and manufacturing. Google Inc. shall have no liability for any error or damages of any kind resulting from the use of this document.

Notices

Table of Contents

List of Tables

List of Figures

1. Introduction

1.1 Scope

This document describes the cryptographic module security policy for the Google Inc. Titan Key cryptographic module with firmware 1.0 (also referred to as the "module" hereafter). It contains specification of the security rules, under which the cryptographic module operates, including the security rules derived from the requirements of the FIPS 140-2 standard.

1.2 Overview

The cryptographic module is a USB 1.1/2.0 compliant Universal 2nd Factor (U2F) token used for two-factor authentication. U2F standardizes how request and response messages are to be sent over the USB transport to U2F key. The U2F protocol is based on a request-response mechanism, where a requester sends a request message to a U2F device, which always results in a response message being sent back from the U2F device to the requester.

After registration, a user can use their Titan Key with an origin-specific key pair across all Google online services. The Titan Key only performs two operations. U2F Register associates a key pair with an origin, google.com here, while U2F Authenticate, verifies that signature with the Titan Key to prove physical possession of the hardware second factor. Then, and only then, is the User able to authenticate to Google services.

The chip-based platform runs a version of the Chrome Embedded Controller, Cr52, which manages all low-level resources, cryptographic algorithms, access control and the life cycle of all keys.

1.3 Glossary

Term	Description
APDU	Application Protocol Data Unit
API	Application Programming Interface
CMVP	Cryptographic Module Validation Program
CSP	Critical Security Parameter
DRBG	Deterministic Random Bit Generator
ECDSA	Elliptic Curve Digital Signature Algorithm
GPIO	General Purpose Input/Output
HMAC	(Keyed-) Hash Message Authentication Code
LED	Light Emitting Diode
KDF	Key-Derivation Function
NDRNG	Non-Deterministic Random Number Generator
PIN	Personal Identification Number
RAM	Random Access Memory
SHA	Secure Hash Algorithm
SRAM	Static Random Access Memory
USB	Universal Serial Bus
U2F	Universal 2nd Factor

Table 1 - Glossary of Terms

2. Security Level

The following table lists the level of validation for each area in FIPS 140-2:

FIPS 140-2 Section Title	Validation Level
Cryptographic Module Specification	1
Cryptographic Module Ports and Interfaces	1
Roles, Services, and Authentication	1
Finite State Model	1
Physical Security	1
Operational Environment	N/A
Cryptographic Key Management	1
Electromagnetic Interference / Electromagnetic Compatibility	1
Self-Tests	1
Design Assurance	1
Mitigation of Other Attacks	N/A
Overall Level	1

Table 2 - Table of Contents

3. Cryptographic Module Specification

3.1 Cryptographic Boundary

The cryptographic boundary is the outer perimeter of the USB PCB. The USB device is a multiple-chip embedded module as defined by FIPS 140-2. The hardware version of the module is 1.0.

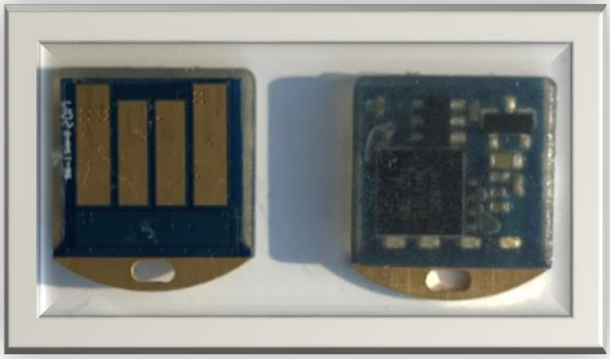

Figure 1 - Titan Key

The physical boundary is depicted in the block diagram below:

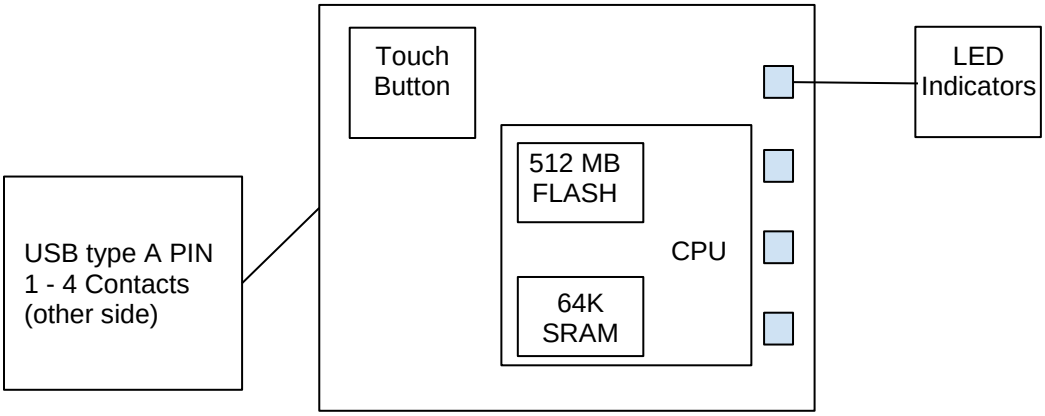

Figure 2 - Titan Key Block Diagram - Top Side

The embedded chip contains the following hardware components:

- General purpose 32-bit processor
- USB 2.0 device controller
- Integrated USB type-A plug
- Non-volatile FLASH memory
- Volatile RAM memory
- 4x LED indicator lights (Only RED and BLUE LEDs active)
- Capacitive touch button.

4. Cryptographic Module Ports and Interfaces

4.1 Physical Interface Description

4.1.1 USB Interfaces

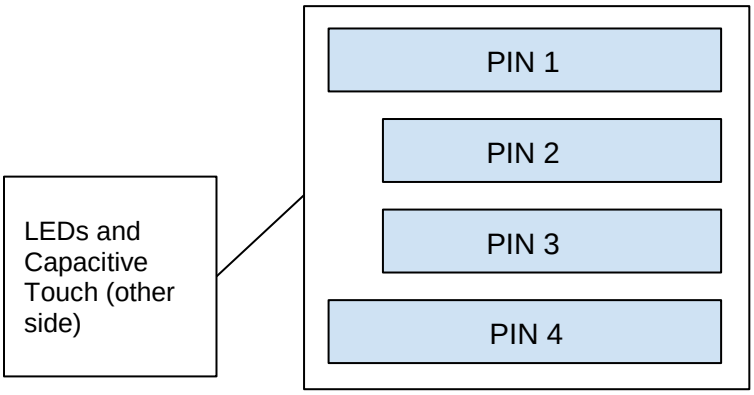

Figure 3 - Titan Key Block Diagram - Bottom Side

Four electrical connections are made between the microcontroller and the USB connector:

- USB PIN 1 - VSS, Ground (reference voltage)
- USB PIN 2 - DM, USB D- connection.
- USB PIN 3 - DP, USB D+ connection.
- USB PIN 4 - VDD, Power supply voltage input.

The above four electronic signals are in full compliance with the USB interface specification. Communications between the host applications and the device is accomplished using an HID device (keyboard) driver that converts back and forth between USB and commands from the microcontroller.

4.1.2 Capacitive Touch button

The capacitive touch button is used to activate the Titan key to register and during authentication. It is located at the opposite end of the PCB assembly from the USB contacts shown in Figure 3.

4.1.3 LED Indicator Lights

LED lights on the module's assembly are located opposite from the USB interface. A Blue LED will illuminate when it is connected to a power source. The Red LED will turn solid red if a FIPS 140-2 related error occurs.

4.2 Logical Interfaces

All communication between the module and a host device is conducted in accordance with the U2F protocol. The U2F protocol is based on a request-response mechanism, where a requester sends a request message to a U2F device, which always results in a response message being sent back from the U2F device to the requester. All request-response messages are framed in ISO7816-4:2005 APDU format. This specifies how to transport the raw message and any error codes if the command failed.

The physical logical interfaces of the module map to the

Physical Port	FIPS 140-2 Logical Interface Mapping	Description
USB PIN2 and PIN3	Data Input Interface	U2F messages passed to the module over the USB interface
USB PIN2 and PIN3	Data Output Interface	U2F response messages passed to the module over the USB interface
USB PIN2 and PIN3 and Capacitive Touch Button	Control Input Interface	Control input passed to the module via U2F messages or the Capacitive touch interface
USB PIN2 and PIN3 and LED Indicator Lights	Status Output Interface	Information returned to the user via the status LEDs
USB PIN4	Power Interface	Power supply voltage 4.75V – 5.25V provided by host device

Table 3 - Physical Port and Logical Interface Mapping

The command protocol and synchronization timing controls, provided in part by way of the clock input, manage the separation of logical interfaces that use the same physical port.

5. Roles, Services and Authentication

5.1 Roles

The module does not provide any identification or authentication for any user that is accessing the device. Since the device does not provide any identification or authentication services, the level of access granted to any functionality of the module is implicitly determined by the service calling the module; the device itself makes no determination about the role itself.

The module supports two independent roles: The Crypto-Officer and the User.

5.1.1 Crypto-Officer Role

The only service allocated to the Crypto-Officer's role is the device firmware update. By design, the role of the Crypto-Officer is limited. They do not, for instance, have any special access to the device by default and a device firmware update overwrites the previous device image.

5.1.2 User Role

The User Role is the main operating role of the module. The module is designed as a complete hardware second factor based on an origin-specific public key linked to a specific web domain (referred to as an origin).

The applications using a module as a second authentication factor are expected to utilize a capacitive touch "proof-of-presence" from the user in order to acknowledge activation of the U2F Authentication functionality (which utilizes all the specified cryptographic algorithms). An individual module (and hence it's User Role) is tied to the user account for a specific origin during a U2F Register event.

The User Role is active so long as the token remains connected and powered.

While it is possible to register a single fob with multiple origins, it is not possible for an origin to become confused about which request to send the fob. Since the origin stores the public, non-secret key handle associated with the account login information, it always sends the correct key handle index to the

correct fob. When this key handle is sent to the fob, the User Role handles the request and activates the proof-of-presence request for direct user confirmation.

5.2 Services

The module provides the following Approved services which utilize algorithms listed in Table 6 and 7:

Service	User	Crypto-Officer
Initialization	X	
Attestation	X	
FIDO U2F: U2F_Register	X	
FIDO U2F: U2F_Authenticate	X	
FIDO U2F: U2F_Version	X	
Signing: Set PIN	X	
Signing: PIN unlock	X	
Signing: Signing	X	
Show Status	X	
Firmware Update		X
On-Demand Self-test	X	
Zeroization	X	

Table 4 - Approved Services and Role allocation

The module provides the following non-Approved services which utilize algorithms listed in Table 8:

Service	User	Crypto Officer
PIN Encryption/Decryption	X	

Table 5 - Non-Approved Services and Role allocation

5.3 Authentication

There is no operator authentication; assumption of role is implicit by the used service(s).

6. Physical Security

The module is a multiple-chip embedded cryptographic module made with production grade components and standard passivation.

7. Operational Environment

The module does not provide a general-purpose operating system.

8. Cryptographic Algorithms and Key Management

8.1 Cryptographic Algorithms

The module implements the following approved algorithms in the firmware:

CAVP Cert #	Algorithm	Sizes	Standard	Mode/Method	Use
4630	AES	128-, 192-, 256-bits	SP 800-38A FIPS 197	CBC, ECB, CMAC	Encryption, Decryption, Decryption, Authentication
1139 1290 (CVL)	ECDSA	P-256	FIPS 186-4	Signature Generation Component, Key Pair Generation, Signature Generation, Signature Verification, Public Key Validation	Digital Signature Services
3065	HMAC		FIPS 198-1	HMAC-SHA-256	Generation, Authentication
3794	SHS		FIPS 180-4	SHA-256	Digital Signature Generation, Digital Signature Verification, non-Digital Signature Applications
1558	DRBG	HMAC-SHA-256	SP 800-90Arev1	HMAC_DRBG	Random Bit Generation

Table 6 - Approved Algorithms

8.1.1 Allowed Algorithms

The module implements the following allowed cryptographic algorithms:

Algorithm	Use
NDRNG	Used only to seed the Approved DRBG
EC Diffie-Hellman	key agreement, key establishment methodology provides 128 bits of encryption strength

Table 7 - Allowed Algorithms

8.1.2 Non-Approved Algorithms

The following non-approved usages are only associated with the legacy PIN encryption/decryption service in Table 5. All algorithms operate by default in approved mode.

Algorithm	Use
KBKDF (non-conformant)	Key-Based Key Derivation Function
AES 128-bits (ECB mode) (non-conformant)	Encryption and Decryption

Table 8 - Non-Approved Algorithms

8.2 Cryptographic Key Management

Keys are generated in the module and loaded during initialization and cannot be changed without zeroizing any keys already loaded into the token. These keys are stored in the module in plain text but cannot be read or exported outside of the token once they have been entered by the Crypto Officer. Only a single set of keys can be loaded in the token for each configuration slot at one time.

The following list of approved keys and CSPs is used by the module. They are generated or inserted as specified and stored within the module as necessary.

Keys and CSPs	Description	Algorithm and Key Size	Generation	Input / Output Method	Storage	Zeroiation
Origin-specific key pair	Per origin asymmetric key pair	P-256 ECDSA key pair	Internally generated by DRBG	Never exits the module	Volatile memory	Session termination
Origin-index obfuscation key	Used to encrypt the origin's key handle	128-bit AES key	Internally generated by DRBG	Never exits the module	Volatile memory	Power cycle
Device attestation key pair	Attests to the authenticity of the device	P-256 ECDSA key pair	Internally generated by DRBG	Public key exits in plaintext	Volatile memory	Power cycle
Per-boot CMAC key	Pin-check key	128-bit, CMAC key	Internally generated by DRBG	Never exits the module	Volatile memory	Power cycle
Host-fob session key pair	EC DH key exchange between host and fob	P-256 EC DH key pair	Internally generated by DRBG	Public key exits in plaintext	Volatile memory	Power cycle
Index key pair(s)	Used to sign arbitrary messages	P-256 ECDSA key pair	Internally generated by DRBG	Public key exits in plaintext	Volatile memory	Power cycle
DRBG State	V and Key values	128-bit	Loaded at the factory; Internally generated using NDRNG	Never exits the module	Volatile memory	Uninstantiation, Power cycle

Table 9 – Approved Keys and CSPs Table

The module implements the following access control policy on keys and CSPs in the module shown in the following table. The Access Policy is noted by R=Read, W=Write and X=Execute.

Module Service	CSP Access	Rights (R/W/X)
Initialization	N/A	
Attestation	Device attestation key pair, DRBG State	RWX
FIDO U2F: U2F Register	Origin-specific key pair, Origin-index obfuscation key, Device attestation key pair, DRBG State	RWX
FIDO U2F: U2F Authenticate	Origin-specific key pair, Origin-index obfuscation key, Device attestation key pair	RWX
FIDO U2F: U2F Version	Device attestation key pair, DRBG State	RX
Signing: Set PIN	Per-boot CMAC key; Host-fob session key, Device attestation key pair, DRBG State	RWX
Signing: PIN Unlock	Index key pair; Host-fob session key, Device attestation key pair	RWX
Signing: Signing	Index key pair, DRBG State	RWX
Show Status	N/A	N/A
Firmware Update[1]	Device attestation key pair	RX
On-Demand Self-test	N/A	N/A
Zeroization	All keys	RW

Table 10 - Approved Service to Key/CSP Mapping

8.3 Key Generation and Entropy

The module employs a NIST SP 800-90A DRBG for key generation. Symmetric keys and seed for asymmetric key pairs are generated as specified in NIST SP 800-133.

The module requests a minimum number of 256-bits of entropy from its NDRNG for use in key generation. The module also implements a factory-derived entropy pool consistent with IG 7.14 2(a) which is filled from an entropic source at manufacturing time prior to shipping. An estimated 512-bits of full entropy is loaded at the factory.

8.4 Zeroization

All private or secret keys are zeroized either at session termination or by power-cycling the module (removing it from the host GPC USB interface and then re-inserting it).

The output data path is provided by the data interfaces and is logically disconnected from processes performing key generation or zeroization. No key information will be output through the data output interface when the module zeroizes keys.

[1] Note: Only validated firmware versions shall be loaded using the firmware update service.

9. Self-tests

FIPS 140-2 requires the module to perform self-tests to ensure the module integrity and the correctness of the cryptographic functionality at start up. Some functions require conditional tests during normal operation of the module.

If any of the tests fail, the module will return an error code and transition to an error state where no functions can be executed. A user can attempt to reset the state by removing the module from the USB port and reinserting it to restart the module. However, the failure of a self-test may require the key to be replaced.

9.1 Power-On Self-Tests

Power-on self-tests are run upon the initialization of the module and do not require operator intervention to run. If any of the tests fail, the module will not initialize. The module will enter an error state and no services can be accessed by the operator.

The module implements the following power-on self-tests:

Type	Test
Integrity Test	• SHA-256 hash over the executable firmware image
Known Answer Test	• AES (Encryption and Decryption. Size 128) • ECDSA (signature generation and verification. Curve: P-256) • HMAC (Generation and verification with SHA-256) • SHS (SHA-256 verified as part the respective HMAC tests) • SP 800-90 HMAC_DRBG

Table 11 - Power-up Self-tests

The module performs all power-on self-tests automatically when it is initialized. All power-on self-tests must be passed before a User/Crypto Officer can perform services. The Power-on self-tests can be run on demand by rebooting the module in FIPS approved Mode of Operation.

9.2 Conditional Self-Tests

Conditional self-tests are test that run during operation of the module. Each module performs the following conditional self-tests:

Type	Test Description
Pair-wise Consistency Test	ECDSA Key Pair generation
Continuous RNG Tests	Performed on NDRNG per IG 9.8
DRBG Health Tests	Performed on DRBG, per SP 800-90A Section 11.3. Required per IG C.1.
Firmware Load Test	ECDSA Signature Verification operation performed prior to a firmware upgrade.

Table 12 - Conditional Self-tests

10. Guidance and Secure Operation

There is no FIPS 140-2 specific guidance required to place the module into its Approved mode of operation. The FIPS 140-2 functional requirements are always invoked.

When the module is inserted into the USB slot of a host device it is powered on its power-up self-tests are executed without any operator intervention. The module enters FIPS mode automatically if the power-up self-test completes successfully. If any of self-tests fail during power-up, the module goes into Error state. The status of the module can be determined by the availability of the module. If the module is available, it has passed all self-tests. If it is unavailable, it is in the error state.

Use of the non-conformant algorithms listed in Table 8 will place the module in a non-approved mode of operation.